Run−Dependent MC Production for Summer 2004

S. D'Auria, A. Dominguez, J. Nielsen, P. Murat, F. Prakoshyn, R. Snihur, R. Tafirout, A. Warburton, D. Waters

- Goals
- Proposal
- Pros & Cons
- Implementation
- Status

Run−Dependent MC : Goals

- Produce Monte Carlo samples that look as much as possible like the data in terms of :
  - detector configuration (e.g. silicon coverage)
  - beam position
  - instantaneous luminosity (additional interactions)
- Have easily extensible Monte Carlo samples such that when more data are added to an analysis, the run range covered by the Monte Carlo can similarly be extended.
- Have a simple way of specifying and reproducing Monte Carlo datasets.
- Make life as simple as possible operationally !

Run−Dependent MC : Proposal

- Generate events corresponding to a **LARGE** (but user specifiable) list of runs. This ensures we map out the luminosity profile of the data properly and don’t bias the samples by picking only long or "golden" runs.
- By default (but user specifiable) generate additional MINBIAS events according to the luminosity for each run.
- By default (but user specifiable) use SVX−GOOD runs, to maximise the usefulness of the MC statistics.
- Make appropriate changes to the way datasets are defined to make samples easily extensible (see following slide).

Example :

run range	threshold	L(pb-1)	L(included)	% included	Nruns
1. 141544-150144	40nb-1	15.8	14.8	93%	118
2. 150145-152625	50	15.9	14.9	93%	88
3. 152636-168892	100	173.4	160.9	93%	396
4. 174996-175292	30	1.7	1.6	93%	18
5. 177214-179056	80	43.3	40.4	93%	112
		250.1	~230	93%	732

choose run ranges corresponding to significant changes in running conditions (especially run size)

must be roughly constant across run ranges to avoid biases

number of runs to generate − still small compared to numbers of events

Run−Dependent MC : Pros & Cons

Pros :

- Monte Carlo looks more like data
- Data/MC scale factors should be closer to unity.
- Stability tests (yields vs. time) can be performed simultaneously on data and Monte Carlo − very useful diagnostic.
- MC calculated efficiencies will have the effect of additional interactions (at least partially) accounted for.

Cons :

- Denominator in (Data/MC) is no longer fixed. Scale factors will be (more) MC sample dependent.
- Slightly more book−keeping required. For example if too many MC files are lost or ignored there is a risk of introducing gaps in MC run−coverage and biases may result.
- Some operational complications such as the need for concatenation of output.

Implementation : Dataset Definitions

- In **mcProduction/book/cdfpewk/** :

```
#-------------------------------------------------------------------------
#  ZEWK0Z: 5.3.2 ppbar->ZZ / Pythia / extra minbias events
#  added   2 events per 1 nb^-1
#-------------------------------------------------------------------------
DSID                    zewk0z
BOOK                    cdfpewk
DSNAME                  pythia_zz_5.3.2
GENERATION_MODE         2
MC_PROCESS_TCL          mc_Pythia_ZZ.tcl
FILTER_TCL              mc_postgenNoFilter.tcl
MINBIAS_TCL             mc_Pythia_Minbias.tcl
NEV_PER_INV_NB          2
NEV_PER_SECTION         5000
FIRST_RUN               141544
LAST_RUN                168899
RUN_LIST                runlist_summer2004
N_SECTIONS              732
#-------------------------------------------------------------------------
```

- **These are the two key parameters**
- **Definition of dataset in terms of N/nb makes it much easier to extend datasets and makes more sense physics wise.**

Implementation : Scripts

- Scripts in **`mcProduction/scripts`** :

```
make_runlist.pl <start_run> <end_run> <lumi_threshold>
```

RUN :	INT LUMI :	INST LUMI :
141572	72.34	1.09e+31
141576	94.88	7.62e+30
141597	159.80	1.14e+31
141598	97.84	7.94e+30
141618	103.80	1.25e+31
141619	82.54	1.00e+31
141660	65.54	1.01e+31
141928	48.30	1.25e+31
141931	139.81	1.01e+31
141984	89.64	1.59e+31

. . . . etc.

$$L = \frac{\text{integrated luminosity}}{\text{livetime}}$$

$$< N_{\text{MINBIAS}} > = L \times \sigma_{\text{INELASTIC}} \times 396\,\text{ns}$$

```
make_joblist.pl <runlist_file> <N/nb> <N/section> [section num]
```

→ Used to calculate number of MC jobs required and by **`MCProd`** to define job properties

Status

- Initial implementation complete.
- Successful test run (dataset "tewk0e" with 1 event/nb; 732 jobs in total).
- Operational problems ironed out (but still some improvements needed to optimise efficiency)
- Have checked to make sure extra MINBIAS events going in as expected :

```
Time per W->enu event on 1.4GHz Athlon CPU :
-------------------------------------------

Inst-L (cm-2 s-1)   :  0        6.9E31

Sim-Time/Evt (sec)  :  6.4      9.6
```

- Ready to generate large datasets.